REMARKS

STATUS OF THE CLAIMS

Claims 53-55 and 58-60 have been pending in the application.

Claims 53-55 and 58-60 are rejected under the judicially created doctrine of obviousness-type double patenting over claim 1 or 2 of US Patent No. 6,557,165.

Claim 60 is rejected under 35 USC 101 for being directed to non-statutory subject matter.

Claims 53-55 and 58-60 are rejected under 35 U.S.C. 102(b) as being anticipated by Maxfield, "Designing With Objects", MachTech, Vol. 7, Issue No. 1, 1991.

According to the foregoing, claim 59 is cancelled without disclaimer or prejudice, the claims are amended, new dependent claim 61 is added, and, thus, pending claims remain for reconsideration which is respectfully requested.

No new matter has been added.

DOUBLE PATENTING REJECTION

Claims 53-55 and 58-60 are rejected under the judicially created doctrine of obviousness-type double patenting over claim 1 or 2 of US Patent No. 6,557,165. However, the claimed embodiments were subject to a USPTO restriction in the parent application, now US Patent no. 8,919,254. In particular, the parent patent application claims were subject to claim Groups I-V restriction, with "component storage medium" claims 53-55 in group IV, which are consonant in scope with claims 53-55 in the present divisional application filed as a result of the restriction requirement, and "component builder apparatus" claims 56-57 in group V, which are consonant in scope with claims 1 and 2 in the divisional application based US Patent No. 6,557,165 (also filed as a result of the restriction requirement) and used as a reference against the present divisional application. According to 35 USC 121 (see MPEP 804.01), it is prohibited to use a patent issuing on an application with respect which a requirement for restriction has been made, or on an application filed as a result of such a requirement, as a reference against any divisional application.

MPEP 804.01(B) suggests that the prohibition against double patenting rejections under 35 USC 121 does not apply if "… claim of the different applications or patent are not consonant

with the restriction requirement made by the examiner, since the claims have been changed in material respects from the claims at the time the requirement was made." However, the original restricted claims 53-55, using claim 53, as amended via a preliminary amendment, as an example, recited:

> 53. (ONCE AMENDED) A component storage medium, comprising storage for [storing] a component which serves as one object in combination with [a predetermined] existing software, said component including a method of issuing an event of the [predetermined] existing software [through a firing by] in response to a message issued in [other] another object.

Currently claim 53 provides:

> 53. (CURRENTLY AMENDED) A computer readable storage medium storing an object oriented programming component accessible by objects in a computer system, comprising:
>
> data related to existing software having a graphical user interface (GUI), the data including GUI identification and GUI event information as GUI data for the existing software with the GUI in the computer system, and
>
> as a component method for the existing software, a program code to control a the computer system according to a process of receiving a message issued in another object, and driving the existing software, including by issuing an a GUI event of the existing software to the existing software, based upon the data related to the GUI data of the existing software, in response to a the received message issued in another object to the existing software,
>
> wherein the data and the component method together in combination with the existing software serve as one object.

It is readily apparent that claim 53 is still consonant in scope to the original claim subject to restriction in the parent, because both claims are directed to "***A computer readable storage medium storing an object oriented programming component accessible by objects in a computer system*** …" and both claims recite "receiving a message issued in another object, and … issuing an a GUI event of the existing software to the existing software, … in response to a the received message" and "wherein the data and the component method together in combination with the existing software serve as one object." Amended claim 53 in relation to the original claim subject to restriction further recites "data related to existing software having a

7

graphical user interface (GUI), the data including GUI identification and GUI event information as GUI data for the existing software with the GUI in the computer system" and "issuing ~~an~~a GUI event of the existing software to the existing software, based upon the ~~data related to the~~ GUI data of the existing software." Similarly, added independent claims 58 and 60 are method claims consonant in scope to the original claims 53 and 55 subject to restriction in the parent. Thus, the obviousness double patenting rejection over patent no. US Patent No. 6,557,165 is prohibited under 35 USC 121, and withdrawal of the same is respectfully requested.

35 USC 101 REJECTION

Claim 60 is rejected under 35 USC 101 for being directed to non-statutory subject matter. The Office Action alleges claim 60 is recites descriptive data/information per se and, thus, not statutory subject matter. According to the foregoing, claim 60 is amended taking into consideration the Examiner comment. Withdrawal of the 35 USC 101 rejection is respectfully requested.

35 USC 102(B) REJECTION

The Examiner maintains from the previous Office Action rejection of claims 53-55 and 58 under 35 USC 102(b) as being anticipated by Maxfield, "Designing With Objects," MachTech, Vol. 7, Issue No. 1, 1991.

The Office Action Response to Arguments alleges "The recitation of the claimed languages remain broad and abstract; therefore, the cited reference reads the claimed limitations:" According to the foregoing, the independent claims 53, 55, 58 and 60 are amended taking into consideration the Examiner comments. For example, the present Application FIG. 101 and page 178, line 4 to page 179, line 14 (claims 53, 60) and FIG. 103, page 180, line 5 to page 183, line 9 (claims 55, 58) support the claim amendments.

If the language of the claims is specifically applied to the Maxfield discussion, a prima facie case of anticipation cannot be established based upon Maxfield, because Maxfield fails to disclose, either expressly or inherently (by not necessarily providing) each and every element set forth in the claimed embodiment, as follows:

The Office Action alleges Maxfield "GetExistingFile" can correspond to the claimed "driving the existing software." However, Maxfield page 5 expressly provides GetExistingFile is a

message to an object, namely the myFileManager object. The claimed embodiment, using claim 53 as an example, does not only recite a message. The language of claim 53 expressly provides "53. (CURRENTLY AMENDED) A computer readable storage medium storing **_an object oriented programming component accessible by objects in a computer system_**, comprising: …" Maxfield page 2, first paragraph, discusses "a message has no software in it … In today's compiled object oriented systems, messages are function calls." Thus, it is readily apparent that Maxfield GetExistingFile message cannot meet, either expressly or inherently, the claimed "as a component method for the existing software, a program code to control a̶the computer system according to a process of receiving a message issued in another object, and driving the existing software, including by issuing an̶a GUI event of the existing software to the existing software, based upon the data related to the GUI data of the existing software, in response to a̶the received message issued in another object to the existing software,." Therefore, the language of the claims provides a method is a program code, which cannot correspond to Maxfield's function call or a message GetExistingFile.

Further, in Maxfield, the message GetExistingFile once received by the object myFileManager causes a call to the routine SFGetFile Dialog, which Diagram 2 represents by a rectangular box. Thus, the SFGetFile is not a method described in the object myFileManager, but SFGetFile is a routine called by the myFileManager object. Thus, GetExistingFile or its function call or message GetExistingFile cannot meet the claimed "as **_a component method for the existing software, a program code_** to control a̶the computer system according to a process of receiving a message issued in another object, and driving the existing software, including by issuing an̶a GUI event of the existing software to the existing software, based upon the data related to the GUI data of the existing software, in response to a̶the received message issued in another object to the existing software," because GetExistingFile is not a method of any object but an external routine called by the method in the myFileManager object.

Further, the Office Action page 3, last paragraph, alleges the rectangular box SFGetFile Dialog meets the claimed "data related to existing software having a graphical user interface (GUI), the data including GUI identification and GUI event information as GUI data for the existing software with the GUI in the computer system." It is understood, the Examiner appears to allege the claimed "existing software" can correspond to MaxField's rectangular boxes, such as SFGetFile Dialog, which are invoked or called by an object, such as myFileManager object.

The SFGetFile Dialog appears to be a system routine, external to the myFileManager object, which is called by the myFileManager object to provide a dialog box for reading an existing file. Maxfield page 3, which is also relied upon by the Office Action page 7 in rejecting independent claim 53, expressly discusses:

> Fourth, a rectangular box is used to indicate an external event which causes some data to flow through the system in the form of messages. This external event may be a mouse click or keyboard entry from the user, a call back routine from a disk write event, or an interrupt from the system time clock. An external event is almost always asynchronous to program execution. The flow of this data is also indicated by labeled arcs.

Thus, Maxfield's rectangular boxes refer to an external event, such as a mouse click or keyboard entry from the user. Although, Maxfield's external event could be software, a conventional object would only use a function call including any parameters to the software as shown in Maxfield Diagram 2, in which myFileManager object calls the SFGetFile Dialog to read an existing file. In contrast, the claimed embodiment of an object provides two items *(1)* "***data related to existing software having a graphical user interface (GUI)**, the data including GUI identification and GUI event information as GUI data for the existing software* with the GUI in the computer system*," and *(2)* "as *a component method* for the existing software*, a program code* to control a~~the~~ computer system according to a process of *receiving a message issued in another object*, and driving the existing software~~, including~~ by *issuing an~~a~~ GUI event of the existing software* to existing software*, based upon the ~~data related to the GUI data of the~~ existing software, in response to a~~the~~ received message ~~issued in another object to the existing software,~~*." In other words, Maxfield is silent on whether the myFileManager object contains the claimed "***data including GUI identification and GUI event information as GUI data**" related to the SFGetFile Dialog, such that Maxfield fails to disclose, either expressly or inherently, the same.

Thus, even if Maxfield's GetExistingFile could correspond to the claimed "existing software having a graphical user interface (GUI)," Maxfield's objects, such as myFileManager object, conventionally call the GetExistingFile, however, in contrast to Maxfield, the language of the claims clearly recites "***an object oriented programming component accessible by objects** in a computer system*" that comprises *(1)* "***data related to existing software having**

a graphical user interface (GUI), the data including GUI identification and GUI event information as GUI data for the existing software with the GUI in the computer system," and *(2)* "as *a component method for the existing software, a program code* to control a~~the~~ computer system according to a process of ***receiving a message issued in another object,*** and driving the existing software~~, including~~ by *issuing* ~~an~~a *GUI event of the existing software to the existing software*, based upon the ~~data related to the~~ GUI *data of the* existing software, in response to a~~the~~ received message* ~~issued in another object to the existing software,~~." Thus, the claimed embodiment relates to an object and could conceptually correspond to Maxfield's objects, such as the myFileManager object. However, Maxfield is silent on how to define an object, such as myFileManger object, but Maxfield only discusses a message flow among objects. Thus, it is readily apparent none of the Maxfield objects expressly disclose the claimed embodiment of "*an object oriented programming component accessible by objects in a computer system*." Further, Maxfield's objects fail to inherently disclose or fails to necessarily include the claimed *(1)* "*data related to existing software having a graphical user interface (GUI), the data including GUI identification and GUI event information as GUI data for the existing software* with the GUI in the computer system," and *(2)* "as *a component method for the existing software, a program code* to control a~~the~~ computer system according to a process of ***receiving a message issued in another object,*** and driving the existing software~~, including~~ by *issuing* ~~an~~a *GUI event of the existing software to the existing software*, based upon the ~~data related to the~~ GUI *data of the* existing software, in response to a~~the~~ received message* ~~issued in another object to the existing software,~~" because Maxfield only discusses a message flow among conventional objects and does not describe details of any object.

Further, in addition to the foregoing patentably distinguishing features, contrast to Maxfield, the claimed embodiment as recited in independent claim 55 and 58, using claim 55 as an example, provides:

> 55. (CURRENTLY AMENDED) A computer readable storage medium to control a data processing system, comprising:
>
> **an object oriented programming component** accessible by objects, comprising:
>
> …

as ***a component method*** <u>***for the existing software***</u>***, a program code*** ~~*to control*~~<u>*controlling*</u> *the data processing system* ~~*according to a process of*~~ <u>*to*</u><u>*:*</u>

…

~~at least one program to control the data processing system according to a process of monitoring the existing software having the graphical user interface for a predetermined~~<u>***monitor a GUI***</u> ***event*** <u>***involved in the existing software***</u><u>*,*</u> **and**

~~to~~ ***issue a message*** <u>***informing another object and/or the component itself of occurrence of the GUI***</u>~~to inform another object of the predetermined~~ *event of the existing software*<u>*.*</u>

…

For example, the present Application FIG. 103 and page 180, line 5+ support claims 55 and 58.

Further, in contrast to Maxfield, claim 60 embodiment provides "***based upon the system information***<u>***, including the GUI data,***</u> ***of the existing software***, *issuing* ~~*an*~~<u>*a GUI*</u> *event* of the existing software, ***issuing a message*** to inform another object <u>and/or the component itself </u>of the <u>GUI </u>event of the existing software, or both, in response to a received message issued by another object to the component method driving the existing software, in response to a predetermined event issued to the existing software, or both."

Further, in contrast to Maxfield, new dependent claims 61 provides "wherein the GUI event to the existing software is issued through a GUI manager of the computer system." It is readily apparent Maxfield is silent on how the Maxfield objects, such as myFileManager, operate, and myFileManager conventional function call to the SFGetFile Dialog differs from the claimed issuing a GUI event. thorough a GUI manager of the computer system.

According to the foregoing, a prima facie case of anticipation cannot be established based upon Maxfield, and withdrawal of the rejection of pending claims and allowance of pending claims is respectfully requested.

CONCLUSION

There being no further outstanding objections or rejections, it is submitted that the application is in condition for allowance. An early action to that effect is courteously solicited.

Finally, if there are any formal matters remaining after this response, the Examiner is requested to telephone the undersigned to attend to these matters.

Respectfully submitted,
STAAS & HALSEY LLP

Date: April 11, 2007

By: / Mehdi D. Sheikerz /
Mehdi D. Sheikerz
Registration No. 41,307

1201 New York Avenue, NW, 7th Floor
Washington, D.C. 20005
Telephone: (202) 434-1500
Facsimile: (202) 434-1501